INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL
ISSN 1841-9836, 13(2), 268-279, April 2018.

EPAK: A Computational Intelligence Model for 2-level Prediction of Stock Indices

L. Tang, H. Pan, Y. Yao

Li Tang
1. School of Management and Economics
University of Electronic Science and Technology of China
Chengdu 611731, China
2. Tianfu College of Southwestern University of Finance and Economics
Chengdu 610052, China
tangli@std.uestc.edu.cn

Heping Pan*
Intelligent Finance Research Center
Chongqing Institute of Finance
Chongqing 400067, China
*Corresponding author: panhp@swingtum.com

Yiyong YAO
Tianfu College of Southwestern University of Finance and Economics
Chengdu 610052, China
yiyongyao@yahoo.com

Abstract: This paper proposes a new computational intelligence model for predicting univariate time series, called EPAK, and a complex prediction model for stock market index synthesizing all the sector index predictions using EPAK as a kernel. The EPAK model uses a complex nonlinear feature extraction procedure integrating a forward rolling Empirical Mode Decomposition (EMD) for financial time series signal analysis and Principal Component Analysis (PCA) for dimension reduction to generate information-rich features as input to a new two-layer K-Nearest Neighbor (KNN) with Affinity Propagation (AP) clustering for prediction via regression. The EPAK model is then used as a kernel for predicting each of all the sector indices of the stock market. The sector indices predictions are then synthesized via weighted average to generate the prediction of the stock market index, yielding a complex prediction model for the stock market index. The EPAK model and the complex prediction model for stock index are tested on real historical financial time series in Chinese stock index including CSI 300 and ten sector indices, with results confirming the effectiveness of the proposed models.
Keywords: empirical mode decomposition, principal component analysis, affinity propagation, k-nearest neighbor, time series, stock index prediction.

1 Introduction

Time series prediction is a practical issue. Especially the financial time series prediction with important economic significance has attracted serious attention from both finance and computer science researchers. A fair large of literatures research on financial prediction methods[23], typically including ARIMA [8,27], ARCH [3,13],GARCH [1,16], chaos-theoretical [17,18], Artificial Neural Network (ANN) [10,20], Support Vector Machine (SVM) [9,19], and K-Nearest Neighbor (KNN) [12,28].

From these equivalent researches, it is realized that the foremost key factor for effective prediction is the feature extraction which should generate essential information effectively. Practically, the financial time series feature extraction is equivalent to the signal analysis. Thus we

can apply a signal processing technique such as the Empirical Mode Decomposition [6] which is an effective method used widely in dealing with nonlinear and non-stationary signals [21,25]. However, the end effect of EMD [15] has not been considered usually in most of the researches except the forward rolling EMD with a sliding window proposed by Zhang and Pan in 2015[26]. Through a forward rolling EMD, the original time series is decomposed into multilevel IMFs with a high dimension, thus the Principal Components Analysis (PCA) [11] can be applied for dimension reduction. Generally, we propose a nonlinear feature extraction procedure integrating a forward rolling EMD and PCA in this paper.

For prediction modeling, KNN is a nonparametric algorithm which can predict via nonlinear regression. However, it should be note that KNN has a large amount of calculations and a large prediction deviation may be occurred when the samples are in disequilibrium. To tackle these issues, this paper proposes a two-layer KNN with Affinity Propagation (AP) clustering. AP is an effective clustering method with sensory signals process and data patterns detection [5]. Integrate the nonlinear feature extraction procedure and a two-lay KNN, this paper proposes a new computational intelligence model for univariate time series prediction called EPAK and construct a complex prediction model for stock market index synthesizing all the sector index predictions applying EPAK as a kernel.

2 An EMD-PCA-AP-KNN (EPAK) financial prediction model

2.1 Frame work of an EPAK financial prediction model

The EPAK model which is a computational intelligence financial prediction model needs to learn, adapt to and evolve along with changing financial situations. In General, we should first define a time frame and assume a historical financial time series long enough exists. In this paper, we focus on a daily time frame thus we can express a financial price time series on day t as

$$X(t) = (O(t), H(t), L(t), C(t), V(t)), \tag{1}$$

where $O(t)$,$H(t)$,$L(t)$,$C(t)$ and $V(t)$ are the open price, high price, low price, close price and volume. In this paper we only consider the close price, so $C(t)$ is set as $X(t)$. We can define a relative return of the price for $X(t)$ as

$$R(t, \lambda) = \frac{X(t) - X(t - \lambda)}{X(t - \lambda)}, \tag{2}$$

where λ is the prediction step length of time series with a basic step length of $\lambda = 1$. Without any other specification, we use $R(t)$ as $R(t, \lambda)$. Therefore, we can express a historical relative return data set as

$$DR(t, T) = (R(t - (T - w) + 1), \cdots, R(t + 1), R(t)), \tag{3}$$

where T is the number of total days and $w << T$ is the width of a sliding window applied for intercepting the historical data set.

In general, an EPAK financial prediction model can be expressed as

$$EPAK : P(t) \Longrightarrow AKNN(k) \Longrightarrow R(t + \lambda), \tag{4}$$

or mathematically

$$EPAK : R(t + \lambda) = AKNN(P(t), k), \tag{5}$$

where AKNN stands for a two-layer KNN with AP, $P(t)$ means the principal components generated by PCA, k is the structural parameter, and $R(t+\lambda)$ is the prediction output. For details, the EPAK model works through two processes, first the feature extraction integrating EMD for signal decomposition of financial time series and PCA for dimension reduction, second the prediction procedure applying a two-layer KNN regression with AP. Thus we can re-express the EPAK model as

$$DR(t,T) => EMD => PCA => P(t) => AP => AKNN(k) => R(t+\lambda), \tag{6}$$

or mathematically

$$R(t,\lambda) = AKNN\{AP[FE(PCA(EMD(DR(t,T))))]\}, \tag{7}$$

where $FE()$ denotes the feature extraction process.

2.2 A nonlinear feature extraction process integrating a forward rolling EMD and PCA

As a foremost key process for financial time series prediction, feature extraction should concentrate essential information from the historical data set and input to the prediction model. The EPAK model encompasses a complex nonlinear feature extraction procedure special for financial time series, integrating a forward rolling EMD and PCA.

A forward rolling EMD for financial time series

To start a feature extraction process, we apply the forward rolling EMD for signal decomposing on the historical data set $DR(i-1,T)$,$i = t,\cdots,t-(T-w)+1$ and make it as the input,$R(t)$,$i = t,\cdots,t-(T-w)+1$ as the output of prediction. Therefore, the input-output data set can be expressed as

$$DP(t,T-w) = \begin{pmatrix} EMD(DR(t-1),T) & \to & R(t) \\ EMD(DR(t-2),T) & \to & R(t-1) \\ \cdots & \cdots & \cdots \\ EMD(DR(t-(T-w)),T) & \to & R(t-(T-w)+1) \end{pmatrix}. \tag{8}$$

After EMD, the original data set has been decomposed to multilevel IMFs which can satisfy two conditions according to IMF definition [6]: 1) for a whole series, the total number of extrema and zero-crossing points should be equal or differ no more than one, 2) for any time period, the mean of the upper envelope and the lower envelope that are formed by local maxima and minima equals zero.

Take the EMD decomposition of $DR(t)$ as an example, the EMD decomposition consists of three parts:

1) Sifting process:

- Calculate all the local maxima and minima of $DR(t)$.
- Generate upper and lower envelopes with maxima and minima by cubic spline, and calculate the mean value

$$m_i(t) = \frac{(ue(t)+le(t))}{2}, \tag{9}$$

where $i = 1,2,\cdots$ indicates the ith-order,$ue(t)$ and $le(t)$ are the upper and lower envelopes.

- Generate a detail component of decomposition

$$h_i(t) = dr(t) - m_i(t), \tag{10}$$

when $i = 1$,$dr(t) = DR(t)$.

2) IMF checking: check whether $h_i(t)$ can satisfy the two conditions of IMF definition:

- If it can, $h_i(t)$ is an IMF, and the residual is

$$r(t) = dr(t) - c(t), \tag{11}$$

$$c(t) = h_i(t). \tag{12}$$

Let $dr(t) = r(t)$ and continue to next sifting.

- If it cannot, let $dr(t) = h_i(t)$ and continue to next sifting by 1).

3) Sifting stop:

- To ensure the instantaneous frequency defined by IMF has sufficient physical significances, a metric SD is defined by Huang [6] to determine whether to stop sifting. SD can be expressed as

$$SD = \sum_{i=1}^{N} [\frac{|h_{i+1}(t) - h_i(t)|^2}{(h_i(t))^2}], \tag{13}$$

when $0.2 < SD < 0.3$, stop sifting.

- If the number of extrema including maxima and minima of $r(t)$ is smaller than two, stop sifting.

- Return all the results as

$$DR(t) = (\sum_{i=1}^{n} c_i) + r, \tag{14}$$

or in more detail

$$DR(t) = \begin{pmatrix} IMF_1(t,w) \\ \vdots \\ IMF_n(t,w) \end{pmatrix} + r. \tag{15}$$

Equation (15) means that EMD can decompose the original data set into n (generally, let $n <= 5$) multilevel IMFs and a residual r,thus Eq.(8) can be rewritten as

$$DP(t, T-w) = \{\boldsymbol{D} \to \boldsymbol{R}\}, \tag{16}$$

$$\boldsymbol{D} = \begin{pmatrix} IMF_1(t-1,w) & \cdots & IMF_n(t-1,w) \\ IMF_1(t-2,w) & \cdots & IMF_n(t-2,w) \\ \vdots & \vdots & \vdots \\ IMF_1(t-(T-w),w) & \cdots & IMF_n(t-(T-w),w) \end{pmatrix}, \tag{17}$$

$$\boldsymbol{R} = \begin{pmatrix} R(t) \\ R(t-1) \\ \vdots \\ R(t-(T-w)+1) \end{pmatrix}. \tag{18}$$

It should be note that each row of $\boldsymbol{D}$ is high dimensional since it consists of multilevel IMFs time series. Therefore, a PCA algorithm can be applied for reducing the high dimension.

PCA for dimension reduction

Compared to other methods of dimension reduction such as Linear Discriminant Analysis (LDA), Locally Linear Embedding (LLE), and Laplacian Eigenmaps (LE), PCA can maintain the information contained in the original data as much as possible after dimension reduction. However, for financial time series prediction, the feature extraction needs to keep the more information of original data the better robustness and prediction accuracy of the model. Therefore, PCA is used for dimension reduction and feature extraction in this paper. PCA reduces data dimension by an orthogonal linear transformation which is actually a singular value decomposition process. Thus we can express a PCA algorithm as three processes:

1) Form a matrix consists of principal components: normalize the high-dimensional matrix $\boldsymbol{D}$ to $\boldsymbol{Y}$

$$\boldsymbol{Y} = (normalization(\boldsymbol{D}))^T . \tag{19}$$

Apply a singular value decomposition of $\boldsymbol{Y}$

$$\boldsymbol{Y} = \boldsymbol{V}\boldsymbol{\Sigma}\boldsymbol{W}^T, \tag{20}$$

where $\boldsymbol{V}$ and $\boldsymbol{W}$ are orthogonal matrices formed by the eigenvectors of $\boldsymbol{Y}\boldsymbol{Y}^T$ and $\boldsymbol{Y}^T\boldsymbol{Y}$ respectively, $\boldsymbol{\Sigma}$ is a nonnegative rectangular diagonal matrix whose left part consists of the eigenvalues $\lambda(i)$,$i = 1, 2, \cdots, p$ of $\boldsymbol{Y}\boldsymbol{Y}^T$. Thus we can generate a transformed matrix $\boldsymbol{P}$ that consists of principal components in turn,

$$\boldsymbol{P} = \boldsymbol{S}^T\boldsymbol{V} = \boldsymbol{W}\boldsymbol{\Sigma}^T\boldsymbol{V}^T\boldsymbol{V} = \boldsymbol{W}\boldsymbol{\Sigma}^T. \tag{21}$$

2) Find a new lower dimension: PCA is used for reducing dimension, thus the dimension of matrix $\boldsymbol{P}$ should be reduced. Actually, the first r in l principal components concentrate the most essential information of matrix $\boldsymbol{P}$. To find the value of r, we can use the Cumulative Contribution Rate (CCR) which is generally required to be more than a threshold (such as 85%) [15] for help,

$$CCR_r = \frac{\sum_{i=1}^{r} \lambda(i)}{\sum_{i=1}^{l} \lambda(i)} > 85\%, \tag{22}$$

where r is the new lower dimension and $r \ll l$.

3) Generate an information-rich matrix: construct a new matrix $\boldsymbol{\Sigma}_r$ as a $r \times l$ matrix by

$$\Sigma_r = \boldsymbol{I}_{r\times l}\Sigma. \tag{23}$$

Then we can form a lower dimensional matrix $\boldsymbol{P}_r$ of $\boldsymbol{P}$

$$\boldsymbol{P}_r = \boldsymbol{W}(\boldsymbol{\Sigma}_r)^T, \tag{24}$$

where $\boldsymbol{P}_r$ is the new low-dimensional matrix which should be input to a two-layer KNN with AP for prediction.

2.3 A two-layer KNN algorithm with AP

KNN is a nonparametric algorithm that can be used for nonlinear regression. To improve the KNN algorithm, this paper proposes a two-layer KNN with AP. We apply AP to transform the feature into clusters as input to a two-layer KNN for prediction.

AP for clusters generation

Assume the feature extracted by a forward rolling EMD and PCA comprises N data points. AP initially regards every data point as a potential cluster center. Then measure the similarity between any data pairs and accumulate evidences for an iterative procedure which can find the final suitable exemplars. For details, we can describe an AP algorithm as three procedures:
1) Construct a similarity matrix: in this paper, we measure the similarity between each data pairs with Euclidean which is one of the classical similarity metric for time series. Thus the similarity can be defined as

$$S_{im}(i,j) = -\|i-j\|^2, \tag{25}$$

where i and j are any data pairs of the N data points. When $j=i$, $S_{im}(i,i)=p(i)$,means the preference of point i can be an suitable exemplar. Therefore, p is set as a preference parameter which can affect the clustering solutions. The similarity matrix $S_{N\times N}$ can be composed of all S_{im} values in order.

2) Find suitable exemplars: an iterative procedure based on $S_{N\times N}$ can be helpful to find suitable exemplars. In this procedure, the evidences of "Responsibility"R and "Availability"A should be accumulated. Assume point c is the exemplar candidate and i is any point. $R(i,c)$ indicates the suitability of c for being the exemplar of i, and $A(i,c)$ indicates the appropriateness of i for choosing c as the exemplar,

$$R(i,c) = S_{im}(i,c) - \max_{j\neq c}\{A(i,j)+S_{im}(i,j)\}, \tag{26}$$

$$A(i,c) = \min\{0, R(c,c) + \sum_{j\neq i,c}\max[0,R(j,c)]\}, \tag{27}$$

When $R(i,c)+A(i,c)$ achieves the maximum, c is the most suitable exemplar of i .

3) Choose an optimal clustering solution: after finding the final exemplars, there are several clustering solutions and the optimal one should be chosen. An effective evaluation of clustering solution is the Silhouette Coefficient which can reflect the separability and compactness between clusters [22]. Assume there are clusters C_i ,$i=1,2,\cdots,n$, SC for any point in cluster C_i can be expressed as

$$SC(x_i) = \frac{\min[a(x_i,C_j)] - b(x_i)}{\max\{b(x_i), \min[a(x_i,C_j)]\}}, \tag{28}$$

where $a(x_i,C_j)$ is the mean dissimilarity between x_i and all the other points in cluster C_j, $j\neq i$,$b(x_i)$ is the mean dissimilarity between x_i and all the other points in cluster C_i . Calculate the mean value of overall SC as

$$SC_m = mean\{\sum_{i=1}^{N} SC(x_i)\}, \tag{29}$$

The higher the SC_m value is, the better the clustering solution [4]. Therefore, we can choose the optimal clustering solution with a highest SC_m .

A two-layer KNN

We propose a two-layer KNN with AP consists of three functions AKNN, AKNN1, and AKNN2. Assume $x(t+\lambda)=DR(t+\lambda,T)$ is the future prediction point and $x(t)=DR(t,T)$ as input. AKNN is equivalent to the input-output of a two-layer KNN. For the first layer which can reduce the computation, AKNN calls AKNN1 to find $x(t)$'s nearest exemplar c_{near} and ascribes $x(t)$ to the same cluster C_{near} . And for the second layer which can ensure equilibrium samples,

AKNN calls AKNN2 to find the k nearest neighbors in C_{near} for prediction. For details, a two-layer KNN can be defined as

Function AKNN inputs $x(t)$ and C_i,$i = 1, 2, \cdots, n$ which are the clusters generated by AP, outputs the prediction $x(t+\lambda) = DR(t+\lambda, T)$,

$$(x(t+\lambda)) = AKNN(x(t), C_i, k). \tag{30}$$

AKNN generates predictions by calling AKNN1 and AKNN2 in turn.

Function AKNN1 inputs $x(t)$ and C_i,$i = 1, 2, \cdots, n$, outputs C_{near} and its exemplar c_{near}

$$(c_{near}, C_{near}) = AKNN1(x(t), C_i, k = 1), \tag{31}$$

where $k = 1$ means c_{near} is $x(t)$'s nearest exemplar. Calculate the similarity between $x(t)$ and each exemplar c_i,$i = 1, 2, \cdots, n$ as

$$S_{im}(x(t), c_i) = -\|x(t) - c_i\|^2. \tag{32}$$

When $S_{im}(x(t), c_{near})$ achieves the maximum, c_{near} is $x(t)$'s nearest exemplar and correspondingly outputs the C_{near}.

Function AKNN2 inputs $x(t)$ and C_{near} ,outputs $x(1+\lambda)$,

$$(x(t+\lambda)) = AKNN2(x(t), C_{near}, k), \tag{33}$$

Calculate the similarity between $x(t)$ and each point x_i in the same cluster C_{near} as

$$S_{im}(x(t), x_i) = -\|x(t) - x_i\|^2. \tag{34}$$

When $S_{im}(x(t), x_j)$,$j = 1, 2, \cdots, k$ achieves the first k $max(S_{im})$ values, x_j,$j = 1, 2, \cdots, k$ are the nearest neighbors of $x(t)$. Thus the prediction $x(t+\lambda)$ can be generated as

$$x(t+\lambda) = \frac{\sum_{j=1}^{k} x_j}{k}. \tag{35}$$

According to Eq. (35), the parameter k is a critical factor which can affect the results, thus find the optimal k value is a practical problem. In general, different specific historical data set has different optimal k value which is usually found by experiments. Therefore, in this paper, we find each optimal k value for each specific model by experiments, setting the original k value as $k = 1$, generating predictions based on nearest neighbors and increasing value with step length $\lambda = 1$. If k value continues to increase three times with no effect on improvement of prediction, we can stop and find the optimal k value makes the best performance of prediction. However, other methods should be studied in future work.

2.4 Three structural parameters of an EPAK financial prediction model

A specific EPAK financial prediction model can be constructed by three structural parameters: 1) λ is the prediction step length, 2) w is the sliding window width used for a forward rolling EMD, and 3) k means the nearest neighbors selected by the second layer of AKNN. Thus Eq. (7) which is a general definition of an EPAK model can be re-expressed as

$$R(t+\lambda) = AKNN\{AP[PCA(EMD(DR(t, w)))], k\}. \tag{36}$$

3 A complex prediction model for stock market index synthesizing all sector indices predictions using EPAK as a kernel

A stock market index can reflect the overall trend of the stock market such as the Chinese benchmarked stock index CSI 300, the Standard and Poor's Composite Index and so on. It is generally calculated by Paasche Idex as

$$I_r = \frac{CAV_r}{b} \times 1000, \tag{37}$$

where CAV_r and b indicate the adjusted market values of the constituent stock (CAV) during the reporting period and on the base date, and b equals a constant that set as a divisor. Moreover,CAV_r can be calculated as [2]

$$CAV_r = \sum_{i=1}^{n} P_i \times AS_i, \tag{38}$$

where n is the number of total constituent stocks, P_i is the price of each constituent stock, and AS_i is the number of adjusted share capital of each constituent stock. Note according to Paasche Index, the circulation or volume of the constituent stock during the report period is set as a weight cw_i, thus AS_i can be calculated as

$$AS_i = TS \times cw_i, \tag{39}$$

where TS is the number of total share capital. Therefore, Eq. (37) can be rewritten as

$$I_r = \frac{1000}{d} \times \sum_{i=1}^{n} TS \times P_i \times cw_i, \tag{40}$$

In general, the constituent stocks of the stock market index are belongs to different industries, accordingly different sector indices can be calculated. Therefore, we can construct a complex prediction model for stock market index synthesizing all sector indices predictions. Applying an EPAK as a kernel, this complex model can be expressed as

$$MIP = \sum_{i=1}^{N} SIP_i \times sw_i, \tag{41}$$

where SIP_i is the prediction of each sector index generated by EPAK, sw_i is the a sector weight which sums the weights of the constituent stocks, and N is the number of total sector indices.

4 Empirical test and results

To test the effectiveness of an EPAK model for predicting a real data set, we do an empirical test on Chinese stock index and collect real historical data set from Wind and China Securities Index Co., LTD (China).

4.1 Performance metrics of a specific EPAK prediction model

For evaluating the performance of a prediction model, some metrics can be applied, generally including Mean Absolute Percentage Error (MAPE), Mean Absolute Difference (MAD), and

Root Mean Square Error (RMSE). However, we apply Hit Rate [14] that can reflect accuracy of the predicted direction as a metric in this paper. It is defined as

$$HitRate = \frac{\sum_{i=1}^{n} h_i}{n}, h_i = \begin{cases} 1, & R_i \times R_p > 0 \\ 0, & R_i \times R_p < 0 \end{cases}, \tag{42}$$

where n indicates the number of samples, R_i and R_p are the real and predicted values.

4.2 A specific EPAK model for predicting CSI 300

Model EPAK_CSI300d1 is proposed to predict the $t+1$ daily return of Chinese benchmarked stock index CSI 300. This model can be expressed as

$$R(t+1) = AKNN\{AP[PCA(EMD(CSI300d1\_DR(t,w)))], k\}. \tag{43}$$

The historical CSI 300 price time series from 4^{th} January 2006 to 29^{th} December 2017 comprises 2917 trading days is used, with the earlier 80% part for in-sample training and the later 20% part for out-of-sample testing. In terms of hit rate, the performance of EPAK_CSI300d1 is shown in Table 1 (Note: The level of IMFs is determined by the EMD decomposition of CSI 300), resulting a highest hit rate of 72.78% with $w = 100$, $n = 3$ and $k = 1$. Therefore, we can say that EPAK_CSI300d1 can predict the $t + 1$ daily return of CSI 300 effectively.

Table 1: Hit rates of EPAK_CSI300d1 for $t + 1$ daily return of CSI 300 prediction

Hit Rate%					
w	k=1	k=2	k=3	k=4	k=5
100	72.78	71.35	68.12	68.29	70.63
150	72.29	70.27	71.19	67.90	67.90
200	70.47	69.91	68.98	69.35	68.05
250	70.85	70.66	67.82	70.10	67.44
300	72.22	70.29	69.13	67.58	69.32
350	68.92	72.66	71.48	68.32	69.31

4.3 A specific complex model for predicting CSI 300 synthesizing ten sector indices predictions using EPAK as a kernel

The constituent stocks of CSI 300 are belongs to ten sectors, thus we construct ten specific EPAK models for predicting the $t + 1$ daily return of these ten sector indices as shown in Table 2 (Note: The data is collected from http://www.csindex.com.cn/). The historical data set for each sector index during the same period as CSI 300's is used. Table 3 shows the test results, where the highest hit rate of each model is selected and the best performance in terms of hit rate is 79.60% with the EPAK model predicting the $t + 1$ daily return of Telecom Svc index. According to Eq. (41), we construct a prediction model for predicting CSI 300 synthesizing ten sector indices predictions as

$$CSI300 = \sum_{i=1}^{10} R_i(t+1) \times sw_i, \tag{44}$$

where $R_i(t + 1)$ is the prediction of each sector index generated by the EPAK prediction model. This model test achieves a hit rate of 73.43% which is higher than the performance of

Table 2: Ten sector indices form CSI 300

Index	Weight (%)	Constituent Number
Energy	2.61	12
Materials	7.04	33
Industrials	13.31	62
Cons Disc	11.92	37
Cons Staples	7.97	11
Health Care	5.15	22
Financials	39.81	75
Info Technology	7.62	30
Telecom Svc	2	7
Utilities	2.57	11

Table 3: Hit rates of EPAK models for daily return of ten sector indices prediction

Index	Hit Rate (%)	Index	Hit Rate (%)
Energy	73.54	Health Care	74.55
Materials	77.58	Financials	69.49
Industrials	78.59	Info Technology	71.52
Cons Disc	78.58	Telecom Svc	79.60
Cons Staples	76.57	Utilities	73.54

EPAK_CSI300d1. This result implies the complex prediction model for prediction the stock market index can improve the effectiveness of the prediction model with more comprehensive information as input.

5 Conclusion

This paper proposes a financial prediction model called EPAK, integrating a nonlinear feature extraction procedure and a two-layer KNN with AP for prediction. The nonlinear feature extraction procedure is adaptable and comprehensive for financial time series analysis and the new two-layer KNN with AP can tackle the main deficiencies of KNN and perform better. Applying the EPAK model as a kernel, we construct a complex model for stock market index prediction which synthesizes the predictions of each sector index of the stock market via weighted average to generate the prediction of the stock market index. Specific EPAK models for univariate time series are implemented for predicting the $t + 1$ daily return of CSI300 and ten sector indices of CSI300, achieving a highest hit rate of 79.60% on Telecom Svc index prediction. Then the predictions of ten sector indices are synthesized via weighted average for generating the prediction of CSI 300, which performs better than the direct prediction of CSI 300.

As a prediction model for financial time series, EPAK comprises two key processes, feature extraction and modeling of prediction which are also the main factors for the performance of prediction. Thus in order to improve the prediction model, we can focus on these two procedures in our future work. For feature extraction, we can advance four aspects on: 1) taking more comprehensive information from different financial markets that interact with each other [7,24], 2) applying other effective nonlinear dimension reduction algorithm integrating other methods which should be more suitable for financial time series, 3) finding a similarity metric to take place of Euclidean measurement which is special for financial time series, 4) improving the prediction modeling, such as apply the Auto Encoder, Random Forest and so on.

Bibliography

[1] Beckers, B.; Herwartz, H.; Seidel, M. (2017); Risk forecasting in (T)GARCH models with uncorrelated dependent innovations, *Quantitative Finance*, 17(1), 121-137, 2017.

[2] China Securities Index Co., LTD (China). CSI 300 index compilation scheme. Shanghai: China Securities Index Co., LTD (China); 2016.

[3] Davidson, J.; Li, X. Y. (2016); Strict stationarity, persistence and volatility forecasting in ARCH process, *Journal of Empirical Finance*, 38, 534-547, 2016.

[4] Dudoit, S.; Fridlyand, J. (2002); A prediction-based resampling method for estimating the number of clusters in a dataset, *Genome Biology*, 3(7),1-21, 2002.

[5] Frey, B. J.; Dueck, D. (2007); Clustering by passing messages between data points, *Science*, 315, 972-976, 2007.

[6] Huang, N. E.; Shen, Z.; Long, S. R. ; et al. (1998); The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis, Proceedings of the Royal Society A:Mathematical, *Physical and Engineering Sciences*, 454, 903-995, 1998.

[7] Hu, C.; Liu, X.; Pan, B.; et al. (2017); Asymmetric Impact of Oil Price Shock on Stock Market in China: A Combination Analysis Based on SVAR Model and NARDL Model, *Emerging Markets Finance and Trade*, 2017 (just-accepted).

[8] Iabal, M.; Naveed, A. (2016); Forecasting inflation: Autoregressive integrated moving average model,*European Scientific Journal*, 12(1), 83-92, 2016.

[9] Jaramillo, J.; Velasquez, J. D.; Franco, C. J. (2017); Research in financial time series forecasting with SVM: Contributions from literature, *IEEE Latin America Transactions*, 15(1),145-153, 2017.

[10] Jena, P. R.; Majhi, R.; Majhi, B. (2015); Development and performance evaluation of a novel knowledge guided artificial neural network (KGANN) model for exchange rate prediction, *Journal of King Saud University-Computer and Information Sciences*, 27(4), 450-457, 2015.

[11] Karl Pearson, F. R. S. (1901); On lines and planes of closest fit to systems of points in space, *Philosophical Magazine*, 2, 559-572, 1901.

[12] Martinez, F.; Frias, M.; Perez, M. (2017); A methodology for applying k-nearest neighbor to time series forecasting, *Artificial Intelligence Review*, 1-19, 2017.

[13] Mokoma, T. J.; Moroke, N. D.(2015); Is the South African exchange rate volatile? Application of the ARCH framework, *Risk Governance and Control: Financial Market & Institutions*, 5(1), 110-122, 2015.

[14] Pan, H. P.; Haidar, I.; Kulkarni, S. (2009); Daily prediction of short-term trends of crude oil prices using neural networks exploiting multimarket dynamics, *Frontiers of Computer Science in China*, 3(2),177-191, 2009.

[15] Pan, H. P.; Ma, Y.; Zhang, C. Z. (2017); FEPA-An integrated computational intelligence model for predicting financial time series, 2017 IEEE/SICE International Symposium on System Integration (SII 2017), 2017 Dec 11-14, Taiwan, China (Accepted).

[16] Pedro, C. S.; Pedro, H. M. (2017); Volatility forecasting via SVR-GARCH with mixture of Gaussian kernels, *Computational Management Science*, 14(2), 179-196, 2017.

[17] Ravi, V. ; Pradeepkumar, D. Deb, ; K. (2017); Financial time series prediction using hybrids of chaos theory, multi-layer perceptron and multi-objective evolutionary algorithms, *Swarm and Evolutionary Computation*, 36, 136-149, 2017.

[18] Rotshtein, A.; Pustylnik, L.; Giat, Y.(2016); Fuzzy logic and chaos theory in time series forecasting, *International Journal of Intelligent Systems*, 31(11), 1056-1071, 2016.

[19] Sermpinis, G.; Stasinakis, C.; Theofilatos, K.; Karathanasopoulos, A. (2015); Modeling, forecasting and trading the EUR exchange rates with hybrid rolling genetic algorithms-Support vector regression forecast combinations, *Harvard Business Review*, 247(3), 831-846, 2015.

[20] Tealab, A.; Hefny, H.; Badr, A. (2017); Forecasting of nonlinear time series using ANN, *Future Computing and Informatics Journal*, 2(1), 39-457, 2017.

[21] Wang, J.; Wang, J. (2017); Forecasting stochastic neural network based on financial empirical mode decomposition, *Neural Networks*, 90, 8-20, 2017.

[22] Wang, K.; Zhang, J.; Li, A.; et al. (2007); Adaptive affinity propagation clustering, *Acta Automatica Sinica*, 33(12), 1242-1246, 2007.

[23] Wen, F.; Gong, X.; Cai, S. (2016); Forecasting the volatility of crude oil futures using HAR-type models with structural breaks,*Energy Economics*, 59, 400-413, 2016.

[24] Wen, F.; Xiao, J.; Huang, C.; et al. (2018); Interaction between oil and US dollar exchange rate: nonlinear causality, time-varying influence and structural breaks in volatility, *Applied Economics*, 50(3), 319-334, 2018.

[25] Yang, H. L.; Lin, H. C. (2017); Applying the hybrid model of EMD, PSR, and ELM to exchange rates forecasting, *Computational Economics*, 49(1), 99-116, 2017.

[26] Zhang, C. Z.; Pan, H. P. (2015); A forecasting model based on forward rolling EMD techniques, *Technical Economics (a Chinese Journal)*, 34(5), 70-76, 2015.

[27] Zhang, G. S.; Zhang, X. D.; Feng, H. Y. (2016); Forecasting financial time series using a methodology based on autoregressive integrated moving average and Taylor expansion,*Expert Systems*, 33(5) , 501-516, 2016.

[28] Zhang, N. N.; Lin, A. J.; Shang, P. J. (2017); Multidimensional k-nearest neighbor model based on EEMD for financial time series forecasting, *Physica A: Statistical Mechanics and Its Applications*, 477, 161-173, 2017.